

39800/DWR/D453

5                   **SYSTEM AND METHOD FOR MANIPULATION AND INTERACTION OF**  
**TIME-BASED MIXED MEDIA FORMATS**

CROSS-REFERENCE TO RELATED APPLICATIONS

10           This application contains subject matter that may be  
considered related to the subject matter disclosed in U.S. patent  
application Attorney Docket No. B600:40288, entitled SYSTEM AND  
METHOD FOR ASSEMBLING DISCRETE DATA FILES INTO AN EXECUTABLE FILE  
AND FOR PROCESSING THE EXECUTABLE FILE; Attorney Docket No.  
5           B600:40289, entitled SYSTEM AND METHOD FOR CONTROLLING  
SYNCHRONIZATION OF A TIME-BASED PRESENTATION AND ITS ASSOCIATED  
ASSETS; and Attorney Docket No. B600:40287, entitled SYSTEM AND  
METHOD FOR TRANSFORMING DATA INTO A TEMPLATE-BASED FORMAT, all  
filed on August 31, 2000, the contents each being hereby expressly  
0           incorporated by reference.

FIELD OF THE INVENTION

          The present invention relates to a system and method for  
collecting and assembling data for subsequent presentation to a  
25          recipient. More particularly, the invention relates to a system  
and method that provides an improved interface to facilitate the  
creation of such a presentation.

BACKGROUND OF THE INVENTION

30          Information is collected and presented to people in many  
different ways. Written text, in the form of books, newspapers,  
and magazines, represent one conventional way of presenting readers  
with information. Electronically, the written text, in the form of

5 text data, may be presented to people over a computer or other similar device. For example, people may access a web site that provides news and other textual information, along with information in other media formats, such as pictures and other images.

10 Another way in which information is presented to people is via a presentation, in which a person communicates such information to a person or group of persons. To assist the presenter in communicating such information, conventionally an overhead projector is used to display a sequence of transparent slides, with each slide typically consisting of text and/or some graphical image.

5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187  
2188  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197  
2198  
2199  
2200  
2201  
2202  
2203  
2204  
2205  
2206

5 are associated sound files, the presentation becomes excessively large, thereby making it difficult to share PowerPoint™ files that include audio clips for one or more of the screen slides. In addition, if an author of a PowerPoint™ presentation wishes to share the presentation with recipients, those recipients must have  
10 either PowerPoint™ or PowerPoint™ Viewer software on their machines in order to view the presentation.

Thus, it would be desirable to have a system and method that facilitate the creation of a multi-media presentation, and that provide an efficient means for a recipient to navigate through the presentation to find desired subject matter. In addition, it would be desirable to have such a system and method that allow an author to record audio clips in an improved manner. Moreover, it would be desirable to create presentations that are compact in electronic file size to facilitate sharing the presentations. And, it would be desirable to have a system that creates a packaged, executable presentation that can be viewed via a conventional web browser, without the need for any additional software. The need also exists to distribute the presentation over a communications network in a streaming media format, thereby mitigating the need for the recipient to download the entire presentation before beginning to view it. The present invention addresses one or more of these desirable features.

#### SUMMARY OF THE INVENTION

30 The present invention provides a system and method for creating a multi-media presentation. In one embodiment, the system and method allow a recipient of the presentation to search through the presentation to retrieve and display desired subject matter. The system preferably includes software that allows an author to

5 record audio clips (e.g., narration) for one or more of the screens  
of the presentation, in any order, and in various manners. In  
addition, the system preferably maintains an internal, searchable  
data file with associated timing information to allow a recipient  
to find desired subject matter within the presentation, and to  
10 display the corresponding screen(s). Moreover, the system may  
transform the plurality of files that comprise the presentation  
into an executable file that can be viewed with a conventional web  
browser.

5 Thus, in one embodiment, the invention is directed to a method  
of preparing a mixed media presentation. Initially, plural sets of  
text data are received for insertion into respective screen slides.  
Audio data for one or more of the screen slides may also be  
received. The text data and associated timing information are  
copied into a data file, and the text and audio data are then  
concatenated together and preferably compressed. Finally, the  
20 screen slides are converted to images, and the images and  
concatenated audio and text data are assembled into an executable  
file, along with additional controlling software.

25 In another embodiment, the invention is directed to a method  
of unpackaging and launching an executable file, including  
providing the executable file including executable code and a  
plurality of blocks of data, running the executable code to  
identify one of the blocks, processing identification information  
in the block to determine the contents of the block, reading the  
30 data in the block and creating a corresponding directory if the  
block is a destination directory block, decompressing the data in  
the block and writing the decompressed data to an appropriate  
directory if the block is a compressed file block, writing the data  
in the block to a temporary directory if the block contains a

5 clean-up program, and saving the information in the block if the  
information contains auto-start path information.

#### DESCRIPTION OF THE DRAWINGS

10 Other features and advantages of the invention will become  
apparent from a description of the figures, in which:

FIG. 1 is a schematic diagram of a system for creating multi-  
media presentations according to one illustrative embodiment of the  
present invention;

FIG. 2 is a flow chart depicting the operational flow of the  
system of FIG. 1 during the creation of a presentation;

FIG. 3 is a flow chart depicting in detail the exportation of  
data into a template-based format according to one illustrative  
embodiment of the invention;

FIG. 4 is a flow chart depicting in detail the assembly of a  
presentation into a single, executable file according to one  
illustrative embodiment of the invention;

FIG. 5 is a flow chart depicting the operational flow of an  
unpackaging process according to one illustrative embodiment of the  
invention;

FIG. 6 is a flow chart depicting the operational flow during  
playback of a presentation created according to the system of FIG.  
1;

FIG. 7 is a flow chart of an event handling process according  
to one illustrative embodiment of the invention; and

FIGS. 8 through 13 are screen shots during creation of a  
multi-media presentation.

#### DETAILED DESCRIPTION OF THE INVENTION

Referring now to FIGS. 1 and 2, there is shown a system 20 for

5 creating multi-media presentations according to one illustrative  
embodiment of the present invention. System 20 includes a user  
interface 22 including an input device 23 and display 28, a  
processor 24, memory 26, and microphone 30. Memory 26 stores  
10 suitable software for creating the multi-media presentations, as is  
described in more detail below.

Input device 23 of user interface 22 may take any suitable  
form, such as a keyboard, keypad, mouse, any other input device, or  
any combination thereof. An author may enter text data through  
user interface 22, or may use the interface to select appropriate  
5 graphical information from a disk storage medium or other source,  
as is described in more detail below.

Processor 24 is connected to user interface 22, and to memory  
26. Processor retrieves the presentation-creating software from  
memory, receives data and control commands from user interface 22,  
20 and displays the presentation information on display 28.

The present invention can be configured to be used,  
independently, by an end-user or, in the alternative, the invention  
can be integrated, as an add-in, into another presentation  
development application. In a preferred embodiment, the system of  
25 the present invention is designed for use in conjunction with  
Microsoft PowerPoint™. It will be understood by those skilled in  
the art that PowerPoint™ is merely one suitable software program  
into which the present invention may be incorporated.

Referring now to FIG. 2, an illustrative method according to  
30 the invention will be described for modifying and preparing an  
existing presentation that consists of multiple digital assets in  
the form of screen slides. Operation begins at step 40, with the  
processor 24 retrieving the presentation-creating software from  
memory 26. At step 42, processor initializes the system 20.

5 Preferably, initialization consists of setting microphone 30 as the currently selected recording object and setting the recording level to one that will result in the recording being at a desirable level, such as 50%.

10 During initialization, processor 24 also preferably resets the size of link sound files. Preferably, processor 24 is programmed to initialize the linked sound files to a relatively large size. In a preferred embodiment, the preset size is 2 megabytes. It will be understood that the file size could be made to be larger or smaller, as necessary.

55 At step 44, system 20 receives an existing presentation, either from an external source, or from memory 26. The presentation consists of a plurality of screen slides arranged in some predetermined order. In one embodiment, the first screen slide of the presentation is presented on display 28. At step 46, the author selects one of the screen slides, for example, by clicking on suitable icons in a tool bar to scroll through the screen slides, through a drop-down menu, or in any other suitable manner.

25 Once the author has selected a particular screen slide, operation proceeds to step 48, and processor 24 receives an audio clip to be linked with that screen slide. A suitable icon is preferably displayed on the screen to alert the author that they can begin speaking the desired audio clip, with the microphone 30 capturing the audio and forwarding the audio data on to processor 30 24. Alternatively, the audio clip can be imported from a file, disk, or the like.

Processor 24 stores the audio data in a suitable temporary file. In addition, processor 24 generates a link between the audio data and the corresponding screen slide, and stores that link,

5 either with the audio clip itself, or in a separate linked file. Alternatively, the audio clip can be stored directly with the screen slide in a slide object, as described in more detail below, thereby obviating the need for any file linking.

10 In another embodiment, the author can progress through all of the slides sequentially, such as if they were making a live presentation, without the need to use the narration capture interface. The narration would be captured automatically along with the slide advance timings. This embodiment is very useful for creating an archive of a live presentation at the time of the live presentation and as a by-product of the live presentation.

At query block 50, processor 24 determines whether there are additional slides for which an author desires to record audio clips. In one illustrative embodiment, processor may query the author whether they wish to record additional audio clips. If so, operator proceeds back to step 46, and the author selects another slide. Alternatively, processor 24 can display the screen slides sequentially, with the author deciding whether to record an audio clip for a particular screen slide when that screen slide is displayed on display 28.

25 If, on the other hand, there are no more audio clips to be recorded, then operation proceeds to step 52, and the author selects one or more of the screen slides for assembling into a final presentation, along with a desired compression format to be employed. Such selection of the slides can be done through a drop-down menu, or by scrolling through the various screen slides and selecting the desired slides, or in any other suitable manner. The selection of the compression format can be done via a drop-down or other suitable menu.



5           Once the author has finished selecting the slides for  
assembly, operation proceeds to step 54, and processor 24 generates  
a playlist object corresponding to the selected slides. The  
playlist object is an intermediate representation of the metadata,  
and contains the semantic and relationship information for the  
10 content, and is a self-contained entity that consists of both data  
and procedures to manipulate the data. The playlist object  
includes a media object to store the audio clips, a screen slide  
object to store the screen images, and a text object to store the  
text contained in the various screen slides. The media, text, and  
5 screen objects also store timing information that defines the  
temporal relationships between the respective types of data, as is  
described in more detail below.

Then, at step 56, processor 24 copies the text from the  
selected screen slides as searchable text data into the text  
object. The text for each slide may be preceded by an appropriate  
header or the like so that a link is maintained between the text  
data and the particular screen slide from which that text data  
originated. At step 58, the individual audio files from each of  
the selected screen slides are extracted from the respective slide  
25 objects and are concatenated into a single audio file which is  
stored in the media object. The single audio file is then  
compressed using the particular compression format previously  
selected by the author, at step 60. Thus, by allowing the author  
to select the compression format and then compressing the audio  
30 file after concatenating the individual audio clips together, the  
author may to some extent control the file size and sound quality.

Alternatively, instead of "physically" concatenating the audio  
files together as described above, a new file may be created that  
maintains plural links to the respective audio files. This is an

5 alternate version of concatenation that may be used in connection with the invention.

At step 62, slide timing information from the selected slides is extracted from each slide object, and the information is stored in a suitable file. For example, each screen slide will have  
10 timing information relating to the start and stop times that the screen slide is to be displayed, which serves to determine the order in which the screen slides are to be displayed.

Then, at step 64, the selected screen slides are saved in a graphics file format, preferably in Graphics Interchange Format ("GIF") format, and stored in the screen slide object. It will be  
15 apparent to those skilled in the art that other suitable graphics file formats may also be used.

At step 66, processor 24 assembles the selected screen slides, in GIF or some other format, with the corresponding audio files, text file, and the file containing the timing information, to  
20 create a single compressed, executable file. The process of forming the single executable file is described in greater detail below in connection with FIG. 4. The executable file may then be forwarded to one or more recipients for subsequent viewing of the presentation. For example, the file can be a Windows® 98 or  
25 Windows® NT standard executable file, as is well known to those skilled in the art. As is well known, an executable file is a binary file containing a program in machine language that is ready to be executed when the file is selected. In that manner, the  
30 executable file may be opened within a suitable web browser or directly in the operating system interface without the need for the presentation software that was used to create the presentation, as is described in greater detail below in connection with FIG. 3.

5           It will be apparent that the system 20 may be used with  
appropriate software to create the entire presentation at one time.  
Thus, rather than retrieving a multi-media presentation, system 20  
can present an author with blank templates, into which the desired  
text and/or graphical data can be entered. Audio clips can then be  
10       recorded for one or more of the created slides, either concurrently  
with the creation of the slide, or after the slides are completed.

Referring to FIG. 3, an export process that is used to process  
data contained in the playlist object is described in detail. The  
export process is designed to transform the data into a template-  
defined data format suitable for display within a browser. The  
export process utilizes a plurality of text templates and slide  
page templates to arrange the meta data from the playlist object  
such that it is in a browser-suitable format, so that the  
presentation can be displayed in a browser without the need for the  
presentation software used to create the presentation. In  
20       addition, executable java scripts and applets are generated and  
inserted into the template, and are run in the browser to allow for  
the presentation to be displayed in the browser.

The export process begins at step 70, with processor 24  
25       retrieving the playlist object with the slides and clips in  
temporal order. At step 72, the export process retrieves a  
template from the set of templates. For example, the template may  
be a text information file that will contain information describing  
how the meta data from the playlist object needs to be formatted  
30       into a format that is suitable for running in the browser. In  
addition, the template contains information relating to the layout  
of the presentation, for example, the relative locations on the  
display of the slides, table of contents, media player controls,

5 search results, and the like. The template also will contain formatting information, for example, text font, size, color, and similar attributes. Moreover, the template also contains references to other files that are used in the display of the presentation.

10 At step 74, the export process processes the command parameters contained in the template to determine what type of file it is, and the destination of the completed file. At step 76, the export process reads the first tag in the template. The tag serves as a text replacement holder. For example, the first tag may instruct the export process to process the table of contents information, the text information, or the slide page information. Within the first tag there are a number of subordinate tags (i.e., there is a hierarchy of inner and outer loops of tags). Thus, where the first tag corresponds to the table of contents, there will be multiple entries to be processed, such as the title of each slide. Then, for the first title, there are plural tags in the template to be replaced with corresponding data from the playlist object. For example, the tags may correspond to character font, size, spacing, positioning, and the like. Thus, each tag is replaced by the corresponding information contained in the playlist object. In the case where the template is a text template, processor 24 retrieves the text-related meta data and inserts that information into the template. Likewise, in the case of a slide page template, the corresponding meta data relating to the slide is retrieved and inserted into the appropriate location of the template based on the tags in the template.

Thus, at step 78, based on the particular tag read by the export process, corresponding meta data is retrieved from the

5 playlist object and inserted into the template, along with  
references to the appropriate files, for example, a slide file or  
the data file containing the actual text data. At query block 80,  
the export process determines whether there are additional tags  
10 remaining in the template to be replaced with information from the  
playlist object. If so, operation proceeds back to step 76.

On the other hand, if all of the tags have been replaced for  
a particular template, operation instead proceeds to query block  
82, and the export process determines whether there are additional  
templates to be processed. If so, operation proceeds back to step  
72. If not, operation proceeds to step 84, and the export process  
5 searches for .tpl files (i.e, template files). For each .tpl file,  
the export process creates a new file for each slide and replaces  
an internal tag with the name of the graphic file. The process  
then terminates.

Thus, by processing the data using the export process and the  
template-defined format, the presentation may be viewed in a  
conventional web browser. Thus, a recipient of the presentation  
need not have PowerPoint™ software in order to view the  
20 presentation.

25 Referring now to FIG. 4, the process of packaging the files  
into a single, executable file is described in detail. Operation  
begins at step 90, with processor 24 receiving input from the  
author regarding packaging information and preferences. For  
example, the author is prompted to input an output file name, the  
30 name of the directory to be packaged (i.e., where the respective  
files are currently stored), a directory name where the unpackaged  
files should be stored, an auto-start file (i.e., the first file to  
be opened when the executable file is selected), and package

5 identification information to uniquely identify the package and its source or origin. At step 92, processor 24 creates and opens an output file into which the single file will be stored.

10 At step 94, executable code is copied to the output file. As is well known in the art, the executable code is the code that is run when an executable file is selected. The executable code controls the unpackaging process, as is described in more detail below in connection with FIG. 5.

At step 96, in the event that package identification information was input by the author, corresponding block identification information and package identification information is written to the output file. The information preferably consists of a starting block flag, block identification information, and the package identification information itself.

Operation then proceeds to step 98, and the destination directory information is stored in the output file, along with a starting block flag and block identification information to identify the contents of the block. Following the identification information, a data string (e.g., a 16-bit string) is written to the output file, which indicates the length of the directory information. And finally, the destination directory information itself is written to the output file.

Then, at step 100, each file in the directory to be packaged is sequentially processed and written to the output file as an individual block. As described above, an author will have previously selected a number of the screen slides to be included in the presentation. Processor 24 accesses the playlist object and retrieves the GIF files for the selected slides from the screen slide object, the single concatenated and compressed audio file

5 from the media object, and the data file containing the corresponding text data from the text object. In addition, processor 24 retrieves the file containing the timing information for the selected slides.

10 At step 100, for each file a starting block flag is written to the output file. File identification information is then stored to identify the file. Next, the string length of the file name is written to the output file, followed by the file name itself. Then, processor 24 determines whether the file is compressed: if not, the file is compressed and stored in a temporary location. Processor 24 next writes information (preferably 32 bits) relating to the size of the compressed file to the output file. Finally, the compressed file is written to the output file, either from the temporary location, or from the originating directory. If a temporary file was created, it is then deleted.

20 Operation then proceeds to query block 102, and processor 24 determines whether the unpackaging directory is a temporary file. If not, operation proceeds to query block 106. If so, operation instead proceeds to step 104, and a clean-up program is retrieved by processor 24 to be included in the output file. The clean-up program is an executable file upon being expanded, and is operative to delete the files contained in a particular temporary file. In this manner, the expanded files contained within the executable file do not permanently occupy memory on the recipient's machine, unless the presentation is intended to be permanently saved on the recipient's machine, in which case a destination directory other than the temporary directory is selected.

Storage of the clean-up program is as follows: first, a starting block flag and clean-up program identification information

5 are written to the output file. Then, the clean-up program is compressed to a temporary location in memory. The length of the compressed program is written to the output file, followed by the copy of the compressed program. The temporary compressed file is then deleted, and operation proceeds to query block 106.

10 At query block 106, processor 24 determines whether one of the files in the bundle was designated as an auto-start file. If not, operation terminates at step 110 with the closing of the output file. On the other hand, if one of the files was designated as an auto-start file, then operation instead proceeds to step 108, and starting block flag and auto-start identification information is written to the output file, followed by a 16-bit string to indicate the length of the auto-start string name, which is followed by the auto-start string itself. Operation then terminates at step 110, and the output file is closed.

20 In an alternative embodiment, the package process inserts a source-identifying block into the package, with such block serving to identify the source of the package. In this manner, the unpackaging process described below can verify the source of the package to ensure that the package does not contain any potentially harmful or offensive data.

25 Referring now to FIG. 5, the unpackaging of the packaged presentation is described in more detail. As is described above, a presentation is packaged into a single, executable file. The executable file may then be transferred to one or more recipients in various ways, either as an email attachment over a communications network, on a disk, or in any other suitable manner.

30 In another embodiment, the executable file is transferred to a host web site, where the file is unpackaged as described below



5 and made available to recipients over the Internet or some other communication network. In that situation, the recipient need not unpack the presentation on their desktop; rather, the unpackaged presentation may be streamed to the recipient slide-by-slide on an on-demand basis.

10 In the case where the executable file is delivered directly to the recipient, operation begins at step 120, with the recipient selecting the executable file, for example, by double clicking on an appropriate icon on the recipient's display. Once the recipient selects the executable file, operation proceeds to step 122, and the executable code in the file automatically runs and scans the data in the output file until it encounters the first starting block flag. Next, the executable code determines the identity of the data contained in the first block, by reviewing the identification information stored in the block during the packaging process.

At query block 124, if the block is determined to be the package identification block, then operation proceeds directly to query block 146 to scan the data for the next block in the file. The package identification block is not processed by the executable code during the unpackaging process. If the block is determined to not be the package identification block, then operation proceeds to query block 126, and the executable code determines whether the block contains unpackaging directory information. If so, then operation proceeds to step 128, and the executable code reads the information contained in the block to determine the full path name of the output directory and subdirectories in which to store the files expanded during the unpackaging process. The code then creates all of the necessary directories and subdirectories into

5 which the expanded files will be stored. Operation then proceeds  
to query block 130, and the executable code determines whether the  
directory into which the files will be saved is a temporary  
directory. If not, operation proceeds to query block 146. If in  
fact the directory is a temporary directory, then operation  
10 proceeds to step 132, and a registry entry is created to control  
the clean-up program to be executed the next time the recipient  
logs in to their machine. Operation then proceeds to query block  
146.

5 If at query block 126 the data block is determined to not be  
a directory information block, then operation proceeds to query  
block 134, and the executable code determines whether the block is  
a compressed file block. If it is, then operation proceeds to step  
136, and the file name for that file is read from the block and  
concatenated with the destination directory. The executable code  
then determines whether a corresponding subdirectory exists and, if  
not, the subdirectory is created and opened. The length of the  
compressed file is determined, and if the data needs to be  
decompressed, it is decompressed and written to the destination  
20 directory. Operation then proceeds to query block 146.

25 If at query block 134 the block is determined to not be a  
compressed file, then at query block 138 the code determines  
whether the block contains the clean-up program. If so, operation  
proceeds to query block 140, and it is then determined whether the  
clean-up program is needed or not, by checking the machine's  
30 temporary directory to determine whether a copy of the program is  
already resident on the machine. If so, operation proceeds to  
query block 146. On the other hand, if there is no resident copy  
of the program, operation instead proceeds to step 142, and the

5 clean-up program is decompressed to an executable file in a temporary directory, such as the Windows temporary directory. Operation then proceeds to query block 146.

10 If the block does not contain the clean-up program, then operation proceeds to step 144, and the executable code determines that the block contains the auto-start file information, and the code saves the path information of the auto-start file for future use. Operation then proceeds to query block 146.

At query block 146, the executable code determines whether there are additional blocks to be unpackaged. If so, the code reads the identification information of the next block at step 148, and operation then proceeds back to query block 124 to determine the block type.

If at query block 146 it is determined that there are no more blocks to be unpackaged, then operation proceeds to query block 150, and the code determines whether there is a file designated as the auto-start file, by checking for auto-start path information. If there is an auto-start file, then operation proceeds to step 152, and the corresponding file is opened to begin the presentation.

25 Where the packaged presentation is transferred to an ASP host, the host is programmed to override the auto-start data and the destination directory information. The host preferably includes codes that investigate the package identification information to ensure that the executable file was generated by a known, trusted source, and not by some unknown entity that might be transmitting a virus or other undesirable content. Once the identity of the author is verified, the package is then unpackaged to a destination directory as determined by the host, and the host stores the

30

5 presentation until a user accesses the host and requests the presentation. The host can then stream the presentation to the user in any suitable, well known manner as described above.

10 Referring now to FIG. 6, playback of the created presentation is described in more detail. Initially, the presentation is obtained by an intended recipient, either on a disk or other storage medium, as an email attachment, or transferred over a computer network, such as over the Internet. Then, at step 200, the recipient opens the file by clicking on a suitable icon representing the presentation, or in any other well-known manner. As described above, when the bundle is extracted by the recipient opening the self-executing file, one of the sub-files is designated as the initial file to be opened, as in conventional in self-executing files. In addition, the extracted files are written to the appropriate destinations for subsequent retrieval during the presentation.

20 At step 202, the presentation is displayed to the recipient, with the slides being sequentially displayed along with any corresponding audio clips for the respective slides. In addition, a table of contents is displayed on the display, and includes the title of each slide in the presentation (FIG. 13). The titles may be selected by the recipient to advance the presentation to the corresponding slide. At query block 204, the recipient's machine (hereinafter "the machine") determines whether the recipient has requested a search for a particular text string within the presentation. In one embodiment, such a request is made by entering the text string in an appropriate box on the screen and then clicking on a corresponding button on the screen (see FIG. 13). If the recipient does not request a search for text, then

5 operation proceeds to query block 205, and the machine determines whether the recipient has made a selection of one of the slide titles in the table of contents. If so, the presentation is advanced to the selected slide, and that slide is displayed along with the corresponding portion of the concatenated audio file, at  
10 step 206. A time-based index into the concatenated audio file is provided, and instructions are transmitted to reposition an audio player to the appropriate point in the audio file based on the time-based relationship between the slide and the audio file. Operation then proceeds back to query block 204. If the recipient  
15 does not select any of the titles in the table of contents, then operation instead proceeds to step 207, and the presentation continues to completion, and operation then terminates.

20 If, on the other hand, the recipient makes a request for a text search, operation proceeds to step 208, and the recipient enters their text string, which is received by system 20. At step 209, the machine accesses the meta data file that was included in the self-executing file and that contains all of the meta data information necessary for playback, including the text that appears on the individual slides. At step 210, the machine compares the  
25 text string with the text contained in the data file. At query block 212, the machine determines whether a match exists. If not, then at step 214 the recipient is notified that there is no match for the entered text string. Operation then proceeds back to step 204, and the recipient may enter another text string to be  
30 searched.

If there is a match between the text string and the text in the data file, operation proceeds to step 216, and the machine retrieves the appropriate GIF file and determines the corresponding

5 position within the single audio file and presents the screen slide  
and corresponding portion of the audio file to the recipient.  
There are many well-known ways in which system may determine the  
appropriate GIF file to retrieve. For example, an association  
table may be maintained to link the text of each slide with a  
10 corresponding GIF file.

It will be apparent that a recipient may request that a search  
be conducted before the presentation begins, during the  
presentation, or after the presentation is completed.

Once the slide and corresponding portion of the audio file are  
presented to the recipient, the presentation may be continued,  
sequentially from the selected slide to the end of the  
presentation, operation may terminate, or operation may proceed  
back to query block 204 to allow the recipient to search for  
another text string.

Referring to FIG. 7, the operational flow of event handling  
software included in one illustrative embodiment of the invention  
is shown in detail. The event handling software controls the  
navigation through a presentation at the recipient's machine. The  
software relies on a set of event data that contains all of the  
information relating to the timing of the presentation. For  
example, the event data includes information concerning the start  
and stop times of each slide page, of each of the clips in a clip  
list, and of each audio clip. In addition, the event data may  
include information concerning when the presentation should  
25 automatically pause or skip to a new position.  
30

Operation of the event handling software begins at step 220,  
and the presentation begins, for example, when the self-executing  
file is opened. The presentation then begins to be displayed, for

5 example, at the beginning of the presentation. At step 222, the recipient's machine is controlled by the event handling software to determine the time of the current position of the presentation. For example, when the presentation is launched from the beginning, the software determines that the time is either at time zero or  
 10 only a few milliseconds. At step 224, the time is compared with the event data for the respective slides, and the slide whose time is either equal to or less than the determined time is selected and displayed in the slide window on the recipient's machine, at step 226.

At step 228, the event handler software calculates a timeout based on the current time of the presentation and the stop time of the slide being displayed. At step 230, the event handler sets a clock to fire an event trigger upon reaching the timeout.

At query block 232, the event handler determines whether the event trigger has fired. If so, then operation proceeds to step 234, and the event trigger initiates a polling process to repeatedly (e.g., every 200 milliseconds) determine the current position of the presentation. At step 236, the current position is compared with the event data for the respective slides. At step  
 25 238, the slide whose time is 1) equal to or 2) less than, and closest in time to, the current time is selected and the slide window is updated with the selected slide. At step 240, the event handler calculates a timeout based on the current time and the stop time of the slide, and resets the clock to fire an event trigger  
 30 upon reaching the new timeout. Operation then proceeds back to query block 232.

On the other hand, if at query block 232 it is determined that the event trigger has not yet fired, operation instead proceeds to

5 query block 242, and the event handler determines whether the presentation has been either paused or stopped by the recipient, for example, by clicking on a pause or stop button, or by selecting another slide for presentation. If not, operation loops back to query block 232. If the presentation has been paused or stopped,  
10 then operation proceeds to step 244, and the presentation is stopped. Also, the event trigger clock is cleared. Operation then proceeds to query block 246, and the event handler determines whether the presentation has been restarted, for example, by the recipient pressing a start button, or repressing the pause button. If the presentation has been restarted, operation proceeds back to step 222 to determine the time of the new position of the presentation.

It will be understood that when the recipient selects a new slide for display, the presentation is automatically restarted at step 246, and operation then proceeds back to step 222.

The above description of the event handler deals primarily with the screen slides themselves. However, it will be apparent to those skilled in the art that the event handler would perform the same functions for synchronizing the display of textual  
25 information, audio clips, and the like.

Referring now to FIGS. 8 through 13, there is shown one illustrative embodiment of various interface screens generated by system 20 to facilitate creation of a multi-media presentation by an author. As shown in FIG. 8, system 20 preferably displays each  
30 of the generated screen slides 21 with the accompanying text for each. In addition, a user interface window 320 is provided to guide an author through the process of creating a multi-media presentation. The user interface navigates the author through the



5 steps of initializing the system 20, recording narration for the  
respective slides 21, previewing a presentation, and packaging the  
final presentation. The user interface 320 includes a Cancel  
button 322, a Back button 324, a Next button 326, and a Finish  
button 328 to allow the author to control navigation through the  
10 process.

FIG. 9 shows a user interface 330 which may be used by an  
author to calibrate the microphone. The calibration of the  
microphone is performed by providing a volume control 332 that can  
be manipulated by the author to adjust the volume of the  
microphone. The range of control spans from 0 to 100%. The screen  
preferably displays the control level at which the microphone is  
set in a display window 334. The level can be increased and  
decreased by manipulating a slide bar 335. To test the volume  
level to determine whether it is acceptable, a control panel 336 is  
provided that enables the author to record and then play back a  
test clip to determine if the volume level of the microphone is  
acceptable. Control panel 336 preferably has a record button 338,  
play button 340 and stop button 342. To test the microphone, the  
author clicks the record button 338 and speaks into the microphone.  
25 When the author is finished recording, the stop button 342 is  
pressed. The author can listen to the recording by clicking the  
play button 340. When the volume level has been set to a desirable  
level, the author can click the NEXT button 326 to continue with  
the creation of a presentation. If, at any time, the author wants  
30 to return to a previous window to change a setting, the author can  
do so by clicking the BACK button 324.

FIG. 10 illustrates a user interface 350 that assists the  
author in narrating a slide. To begin recording on a particular

5 slide, RECORD button 352 is clicked. The author can stop the  
recording at anytime by clicking on STOP button 354. The author  
can also pause the recording by pressing PAUSE button 356. The  
author can play back the recording by clicking on PLAY button 358  
to ensure that the audio clip is audible and clear. If the content  
10 is not as desired, the author can override the previous audio clip  
by recording over it. In addition, interface 350 includes Previous  
and Next Slide Buttons 357 and 359, which allow the author to  
navigate through the respective slides 21. Thus, the present  
system allows the author to record the slides out of order, giving  
the author greater independence in working on the slides in the  
order desired by the author and, not necessarily, in the order that  
the slides appear in the presentation material.

When finished narrating a slide, the author can proceed to the  
next slide by clicking on NEXT slide button 359, or to a previous  
slide by clicking on PREVIOUS slide button 357. The activation of  
either of those buttons will automatically terminate the narration  
for that slide. Thus, it will be apparent that user interface 350  
allows the author to record narration for the respective slides in  
any order. The audio for each slide is independent of the other  
25 slides, and thus, the audio for the slides can be recorded in an  
order independent of the order of the slides.

The interface 350 preferably includes a slide time meter 364  
that displays the length of the audio for each slide and a total  
time meter 366 that displays the length of the audio for the entire  
30 presentation. This allows the author to keep track of the length  
of the entire presentation as the author is recording the audio for  
each slide.

5 In addition to providing information regarding the length of the audio recordings on interface 350, the length of the various audio recordings are also provided as time meter displays 367 under each slide. This enables the author to view the audio recording length for all of the slides simultaneously.

10 In one embodiment, system 20 requires that narration be recorded for each slide, with the length of the recording determining the length of time for which the slide will be displayed. Alternatively, if narration is not recorded for a particular slide or slides, a default display time may be assigned to that slide, such as 4 seconds or some other amount of time. Or, system 20 may query the author to enter a default time for a particular slide for which no narration has been recorded.

FIG. 11 shows a user interface 360 that allows an author to preview and/or package a finished presentation. Interface 360 includes a Preview button 362, which if clicked causes system 20 to launch the presentation immediately, so as to allow the author to preview the presentation before completion. The presentation material can be packaged so as to be optimized for sound quality or optimized for size. The author makes their selection by clicking on one of two windows 365 and 367. Clicking on the preview button causes the processor 24 to carry out the concatenating, compressing, and export processes so as to have the data in a format suitable for presentation within the web browser. In addition, the preview function causes the processor to launch the auto-start file in the web browser automatically, as would be done by the unpackaging process described above.

FIG. 12 depicts a user interface 370 to allow the author to select a file name under which the presentation will be stored. In

5 a preferred embodiment, optimizing the presentation for size provides a compression of about 6500 bits per second, whereas optimizing for sound quality provides a compression of about 8500 bits per second. In the embodiment shown in FIGS. 9 and 10, the user interfaces 360 and 370 only allow the author two choices for optimization, namely, optimization for sound quality and optimization for size. The system, however, can be adapted to provide additional optimization choices to the author. For instance, if authors desire a different level of audio, e.g., audio at a high bandwidth to facilitate high quality CD recording, the system can be adapted to provide additional optimization choices to the author. The optimization for sound quality and size are preferable for presentation materials that mainly contain audio consisting of the spoken word. In the case of audio containing spoken word, the frequency response is small and telephone-like, monaural quality is the audio level that is required to be provided by the system.

User interface 370 assists the author in saving the presentation material. In one illustrative embodiment, system 20 keeps track of the last used directory 374 and displays the directory name in the "save file as" window 372. That directory is concatenated with the current name 376 of the presentation material to create the file name for the presentation. For instance, user interface 370 displays a directory of "D:\MyDocuments\" and a file name of "testing of hotfoot.exe." The presentation material is thus saved in the specified directory under the specified file name. User interface also includes a Browse button 378 to allow an author to select another path in which to store the presentation.

5 In yet another embodiment, system 20 inserts a default directory into window 372, rather than the last-used directory.

As is described above, the creation of the playlist object allows the system of the present invention to be compatible with numerous other applications because the playlist object simplifies, generalizes, and abstracts the process of data storage, post-processing, and transmission. Thus, the playlist can be reused in other applications, while the playlist ensures the referential integrity, provides object modeling, provides consistency between what is done in the present system with other applications, which allows efficient and compatible data sharing between different applications.

The system loops through each slide, extracts the text of the slide, and removes the megaphone object from each slide and exports it as a .gif file. The exportation of the slide object as a .gif file can be done by using Microsoft PowerPoint™. Auto-numbering is automatically turned off by the system so as not to get a "1" at the bottom of each page. The duration of the audio file for each file is measured and, if the slide has no audio duration, a duration of four seconds is assigned. The reason for assigning a four second duration is that the recipient's application is responsible for advancing the slides. If there is no audio recorded for the slide, the slide will be shown for four seconds and then is automatically advanced to the next slide.

The corresponding audio clips for the selected slides are also retrieved and saved as .wav files. The .wav files are concatenated and integrated together. The .wav files can also be converted to other digital media continuous stream formats, such as MP3. It will be apparent to those skilled in the art that by concatenating

5 the files together, prior to encoding into another digital sound  
format, the notion of independent audio linked to slides is  
transformed into a coherent and unchangeable presentation format.  
The coherent format allows the recipient to jump from slide to  
slide randomly and out of order but does not allow the recipient to  
10 modify or change the audio or the slides. Therefore, the intention  
of the publisher is preserved.

In one illustrative embodiment, the .wav file is converted to  
a Windows Media file and the bit rate is set to the bit rate  
previously determined by choosing optimization for size or sound  
quality. The Windows Media file is a single media file that can be  
5 attached to the playlist object.

The author has the option of choosing which slides will be  
included in the presentation package, with such selections being  
done in any suitable manner, such as by clicking on a window next  
to each slide, through a drop-down menu, or in any other suitable  
manner. For instance, the author can chose slides 1, 3 and 7 and  
those become packaged in the presentation. Or the author can  
unselect 3 and select another slide, for instance slide 8. The  
fact that the audio is tied to a slide, as opposed to across the  
entire presentation, allows the author to chose the order and the  
slides to be included in the presentation. System 20 extracts the  
20 necessary information from the selected slides only when packaging  
the presentation.

The packaged presentation is then subjected to the above-  
described export process, in which the necessary information is  
30 extracted from the playlist object and put into a template-defined  
format suitable for display within a browser. In one embodiment,  
system 20 stores the title of each slide in a suitable file for

5 creating the table of contents, and strips all of the text from  
each slide and stores the text in another file, as is described  
above. The information proceeds to the packaging process which, as  
described in detail above, takes the files and subdirectories,  
including the media file and the slides, and creates an executable  
10 file.

5 The packaging process gathers a relatively large number of  
files, for example as many as 30 to 40 individual files, that are  
created by system 20 when a slide presentation is created. There  
may also be other files, such as external files, that also need to  
be included in a presentation. The packaging process gathers the  
external files along with the presentation files and creates a  
single, simplified package. In a preferred embodiment, the  
packaging and unpackaging functions are completed without  
interfacing with the author. One of the files in the package is  
designated as the file to be opened when the package is extracted.  
A marker is also placed in the executable file that identifies the  
file as one that is compatible with the application of the present  
system.

25 Referring now to FIG. 13, there is shown a portion of the  
presentation, for example when an author has selected the Preview  
option. The presentation includes a table of contents 400 that  
includes the title for each of the slides. Each title may be  
clicked on to immediately display the corresponding slide. In  
addition, the presentation displays one of the slides 21.  
30 Moreover, the presentation includes a window 402 into which the  
recipient may enter a text string to be searched for. A Search  
button 404 is provided and may be selected by the recipient to  
begin a search for text, as is described above in more detail. The

5 search results are displayed in a portion of the screen 406. In  
one embodiment, if there is a match, the slide that contains the  
matched text is automatically retrieved and displayed, along with  
the corresponding audio clip. Alternatively, the results may be  
10 displayed for the recipient, with the recipient then selecting one  
of the slides for display. The display preferably also includes a  
Play button 408, Pause button 410, and running indicator bar 412 to  
indicate the current state of the presentation.

While the above description has focused primarily on a  
presentation consisting of screen slides and corresponding audio,  
it will be readily understood by those having ordinary skill in the  
art that the various aspects of the present invention have utility  
in connection with other data-presenting formats. For example, the  
present invention may be used to export, package, unpackage, and  
display presentations consisting of spread sheets and corresponding  
audio clips for one or more of the respective cells in the spread  
sheet, word processing documents with corresponding audio clips for  
the various pages of the document, charts, screen capture  
scenarios, and the like. It will be understood that the various  
aspects of the invention, including the packaging process, export  
process, unpackaging process, and event handling process, have  
25 utility in connection with various different types of information,  
and that the screen slide presentation described herein is but one  
illustrative embodiment of the utility of the invention. Thus, the  
export process, packaging process, unpackaging process, and event  
30 handling process can each be used in connection with various types  
of information.

By way of example, in the case of a spreadsheet, the present  
invention may be used to add audio clips (e.g., voice comments) to



5 particular cells within the spreadsheet, in a similar manner to the  
audio clips being associated with the respective screen slides.  
The invention will concatenate the audio clips into a file,  
compress the file, and assemble the compressed file, spreadsheet  
graphics file, and the other files described above into a single,  
10 executable file.

In addition, a word processing document can be associated with  
one or more audio clips, wherein the audio clips are linked to  
particular pages, chapters, paragraphs, and the like, of the  
document. The export process, packaging process, and unpackaging  
process are carried out in much the same way as in the case of the  
5 screen slide presentation.

As used herein, the term "digital asset" is defined as a  
collection of data that is presented to a viewer, such as a screen  
slide, a video clip, an audio clip, a spreadsheet, a word  
processing document, and the like.  
20

As described above, in the case of a presentation consisting  
of plural screen slides, the text from each screen slide is  
preferably extracted and stored in a data file, with such data  
being available for searching during subsequent presentation.

25 Where the invention is dealing with other types of digital assets,  
some other type of data may be extracted from the respective assets  
for use in intelligently navigating through the presentation. For  
example, in the case of a video signal, closed captioning  
information may be extracted from the video and stored in the data  
file. Alternatively, selected video frames may be extracted and  
30 stored, such as transitory frames or other important frames.  
Moreover, in the case of audio data, key words may be extracted  
from the audio and stored in the data file.

5           In addition, while the above description focuses primarily on  
audio clips being linked to respective digital assets (e.g., screen  
slides, video clips, and the like), the audio clips can be replaced  
with any continuous stream media format, such as video, audio and  
video, animations, telemetry, and the like. Thus, the invention  
10 has utility with any continuous stream media format, and it will be  
understood by those skilled in the art that audio clips are but one  
example thereof.

From the foregoing, it will be apparent to those skilled in  
the art that the system and method of the present invention provide  
for creating a multi-media presentation in a user-friendly,  
efficient manner. The system also provides a presentation that can  
be searched to arrive at desired subject matter.

While the above description contains many specific features of  
the invention, these should not be construed as limitations on the  
scope of the invention, but rather as exemplary embodiments  
thereof. Many other variations are possible. Accordingly, the  
scope of the invention should be determined not by the embodiments  
illustrated, but by the appended claims and their legal  
equivalents.